AD-A264 181

AR-008-257

THE WAKE SOFTWARE SUITE: USER'S GUIDE

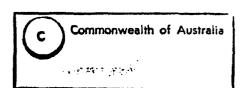
(g)

M.A. CARROLL

MRL-GD-0050 JANUARY 1993 S DTIC ELECTE MAY 1 3 1993 D

APPROVED

FOR PUBLIC RELEASE



93-10324

MATERIALS RESEARCH LABORATORY

DSTO

93 5 11 090

The Wake Software Suite — A Program to Predict the Internal Waves Generated by a Moving Subsurface Body in a Density Stratified Ocean: User's Guide

M.A. Carroll

MRL General Document MRL-GD-0050

Abstract

This report provides details of the requirements needed to run the computer program WAKE. WAKE was developed during the period February-March 1992 as a result of a research project which was undertaken within DSTO Salisbury. WAKE calculates the internal wave velocity field generated by a moving subsurface body in a density stratified ocean.

DSTO MATERIALS RESEARCH LABORATORY

DTIC QUALITY INSPECTED 1

Accesion For

NTIS CRA&I
DTIC TAB
Unannounced
Justification

By
Distribution /

Availability Codes

Oist Special

P. \

Published on behalf of

M.A. Carroll
EBOR Computing

by

Materials Research Laboratory Cordite Avenue, Maribyrnong Victoria, 3032 Australia

Telephone: (03) 246 8111 Fax: (03) 246 8999

© Commonwealth of Australia 1993

AR No. 008-257

THE WAKE SOFTWARE SUITE - A PROGRAM TO PREDICT THE INTERNAL WAVES GENERATED BY A MOVING SUBSURFACE BODY IN A DENSITY STRATIFIED OCEAN: USER'S GUIDE

by
M.A. Carroll
EBOR Computing
in collaboration with E.O. Tuck - University of Adelaide and the following DSTO staff: R. Webster, G. Furnell, A. Legg.

Abstract

This report provides details of the requirements needed to run the computer program WAKE. WAKE was developed during the period February-March 1992 as a result of a research project which was undertaken within DSTO-Salisbury. WAKE calculates the internal wave velocity field generated by a moving subsurface body in a density stratified ocean.

INTERNAL WAVES IN SUBMARINE WAKES

M.A. Carroll

EBOR COMPUTING

Defence Science & Technology Organisation Salisbury, S.A. 5108

Table of Contents

1. Introduction	
2. Running Wake	4
2.1. Parameter File	
2.2. Density Profile	
3. Output	5
3.1. Velocity Component Output File	
3.2. Integration File	
4. MATLAB	1.4
4.2. 2-D Wave Plots	
4.3. Matrix Intensity Plots	
4.4. Contour and Field Flow Plots	
4.4.1. Contour Plots	
4.4.2. Field Flow Plots	
4.5. Printing	
5. Compiling	21
6. Program Description	23
6.1. Stage 1: Initialisation	
6.2. Stage 2: Binding the roots of D(0,k)	
6.3. Stage 3 : Solving the ODE	
6.3.1. Amplitude Functions $A(\theta)$, $B(\theta)$	
6.4. Stage 4: Integrands and Integration	
7. Testing	21
/· Iosuig	31
8. Error Messages	37
9. Related Documents	40

1. Introduction

From February - March, 1992 Professor Ernie Tuck from the Applied Mathematics Department of Adelaide University undertook an investigation for DSTO of Submarine Wakes. The emphasis of the investigation was to compute the internal waves generated by the passage of the submarine through a stratified ocean. Ebor Computing was engaged to provide software support to Professor Ernie Tuck and DSTO. This support consisted of developing software to model these internal waves and produce output in a form so that it may be studied or used by graphics packages such as MATLAB and PLOTZ to display images of the wakes.

Figure 1 is a diagram showing the overall structure of the software developed.

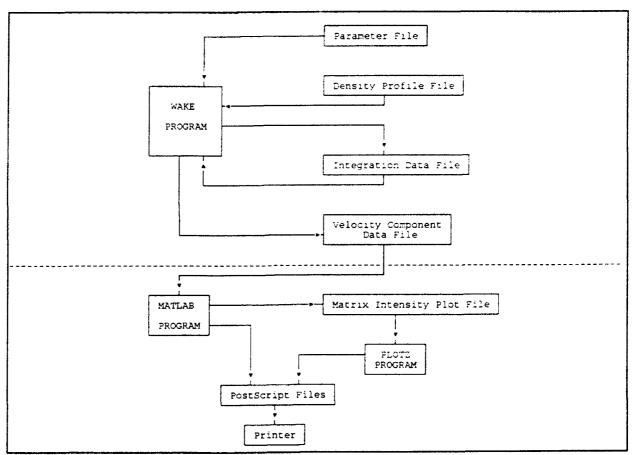


Figure 1 : System Diagram

This document describes the program developed by Professor Ernie Tuck, Dr. Graham Furnell, Mr. Tony Legg from DSTO and Mr. Michael Carroll of Ebor Computing.

2. Running Wake

The program was developed on a SparcStation using MATLAB and PLOTZ to generate plots of the wakes. To run the program you need to copy all the source, parameter and density p ofile files into your directory and compile them, see Chapter 5 on Compiling. You will also need access to the IMSL Library Routines (Fortran Version). Once the program has been compiled the program is run as follows:

> wake

2.1. Parameter File

The user is not prompted for any inputs, all inputs to the program are contained in the parameter file WAKE.PAR. Figure 2 shows a sample parameter file:

NB: The Parameter file must have the following format:

3 Header/Comment lines :

23 Parameter Lines:

Gives information such as file purpose and date.

First 71 Characters contain a comment on the

parameter, the rest of line contains the actual

parameter.

```
This file is read by WAKE and contains initial values for some of WAKE's
options.
                                              Michael Carroll
                                                                23/4/92
NUMTHETA - The number of theta values to calculate data for (<=30) : 30
          - The mode to calculate data for
MODE
THETAMAX - The maximum theta value allowed (<= Pi/2)
                                                                    : 1.56
SUBDTH
         - The depth of the submarine (in metres)
                                                                    : 50.
SUBSPD
         - The speed of the submarine (in metres/second)
                                                                   : 2.5
SUBLEN
         - The length of the submarine (in metres)
                                                                   : 100.
SUBRAD
         - The radius of the submarine (in metres)
         - The starting point for the X-Grid (in metres)
                                                                   : 1000.
XSTART
XEND
         - The end point for the X-Grid (in metres)
                                                                   : 1000.
         - The step size in X (in metres)
XSTEP
                                                                   : 0.
                                                                   : 0.
YSTART
         - The starting point for the Y-Grid (in metres)
         - The end point for the Y-Grid (in metres)
YEND
                                                                   : 2000.
         - The step size in Y (in metres)
YSTEP
                                                                   : 10.
ZSTART
         - The starting point for the Z-Grid (in metres)
                                                                   : 0.
                                                                   : 480.
ZEND
         - The end point for the Z-Grid (in metres)
ZSTEP
         - The step size in Z (in metres)
                                                                   : 10.
         - The depth of the ocean
DEPTH
                                                                   : 480.
DENFIL
         - Name of the file containing the density profile info.
                                                                   : dawson.pro
NUMDEN
         - Number of points in the density profile, DENFIL
                                                                   : 97
                                                                  : debug.dat
DBGFIL
         - Name of the file to write the debugging information to
         - Name of file to receive output velocity component data
VELFIL
                                                                   : arrow.dat
INTFIL
        - Name of the file to receive integration data
                                                                   : inc.dat
COMPU
         - Compute the U component velocity ?? (Y) or (N)
                                                                   : N
         - Compute the V component velocity ?? (Y) or (N)
COMPV
                                                                   : Y
COMPW
         - Compute the W component velocity ?? (Y) or (N)
```

Figure 2 : Sample Parameter File

The parameters read in convey the following information:

NUMTHETA: An integer in the range 2..30. NUMTHETA describes the number of

theta values to divide the interval θ_{min} .. $\pi/2$ into. The amplitude

functions are then calculated at those theta values.

MODE: An integer in the range 1..10. MODE specifies the internal wave

mode that we wish to calculate the data for.

MAXTHETA: The maximum theta value allowed. MAXTHETA $< \pi/2$.

SUBDTH: A real in the range 0. Depth of the Ocean. SUBDTH specifies the

depth of the submarine in metres. The Depth of the Ocean is the

input parameter DEPTH.

SUBSPD: A real in the range 0. Maximum Speed. SUBSPD specifies the speed

of the submarine in metres/second. Maximum speed is set at 30 m/s.

SUBLEN: A real. SUBSPD specifies the length of the submarine in metres.

SUBRAD: A real. SUBRAD specifies the radius of the submarine in metres.

XSTART: A real. XSTART specifies the point in the X-Plane at which the

integrations will be started from.

XEND: A real. XEND specifies the point in the X-Plane at which the

integrations will end. XEND > XSTART > 0

XSTEP: A real. XSTEP specifies the size of the X-Grid (i.e. the spacing

between consecutive points in the X-Plane at which an integration

will be calculated. XSTEP > 0

YSTART: A real. YSTART specifies the point in the Y-Plane at which the

integrations will be started from.

YEND: A real. YEND specifies the point in the Y-Plane at which the

integrations will end. YEND > YSTART > 0

YSTEP: A real. YSTEP specifies the size of the Y-Grid (i.e. the spacing

between consecutive points in the Y-Plane at which an integration

will be calculated. YSTEP > 0

ZSTART:

A real. ZSTART specifies the point in the Z-Plane at which the

integrations will be started from. (ie first depth position) ZSTART must be a multiple of : DEPTH / ((NUMDEN-1)/2)

ZEND:

A real. ZEND specifies the point in the Z-Plane at which the

integrations will end (ie last depth position).

ZEND must be a multiple of: DEPTH / ((NUMDEN-1)/2)

ZSTEP:

A real. ZSTEP specifies the size of the Z-Grid (i.e. the spacing between consecutive points in the Z-Plane at which an integration

will be calculated.

ZSTEP must be a multiple of : DEPTH / ((NUMDEN-1)/2)

DEPTH:

The depth of the ocean as described by the density profile. DEPTH >C

DENFIL:

A character string containing the name of the file which holds the

density profile.

DENSITY PROFILE MUST HAVE AN ODD NUMBER OF POINTS

NUMBEN:

The number of points in the density profile, DENFIL.

DBGFIL:

A character string containing the name of the file to receive the

debugging information.

VELFIL:

A character string containing the name of the file to receive the

velocity component data.

INTFIL:

A character string containing the name of the file to receive the

integration data.

COMPU:

A character indicating whether the U component velocity should be calculated. "Y" or "y" indicates "YES!, Calculate the U component" "N" or "n" indicates "No!, Calculation of the U component is not

required".

COMPV:

A character indicating whether the V component velocity should be calculated. "Y" or "y" indicates "YES!, Calculate the V component" "N" or "n" indicates "No!, Calculation of the V component is not required".

COMPW:

A character indicating whether the W component velocity should be calculated. "Y" or "y" indicates "YES!, Calculate the W component" "N" or "n" indicates "No!, Calculation of the W component is not

required".

2.2. Density Profile

The density profile, $\rho(z)$ is contained in the file specified in the parameter DENFIL. The values of $\rho(z)$ are used to calculate $\mu(z)$ according to

$$\mu(z) = -\frac{\rho'(z)}{\rho(z)}$$

where z is the dep_n ranging from 0 (ie the ocean surface) to DEPTH (ie the ocean floor) (NB: $\rho'(z)$ is approximated in the program by a finite difference method)

Figure 3 shows a sample density profile.

NB: The Density Profile must have the following format:

4 Header/Comment lines:

Gives information such as the files purpose and date.

"NUMDEN" Data Lines: These lines contain the density data. The first line

contains the density at the Ocean Floor and the NUMDENth Line contains the density of the Ocean Surface. NUMDEN has to be odd and there should

be no blank lines at the end of the file

The profile shown is taken from T.W. Dawson's DREP Internal Wave Normal Mode - Theoretical Background Report, 1988. [DREP TM88-7]

```
Dawson Density Profile.
                                                                      Tony Legg 23/3/92 Interpolated at
every 5 m.
Depth of the Ocean = 480 m.
Number of Points = 97
1.0029176
1.0029131
1.0029087
1.0029042
1.0028996
1.0028951
1.0028907
1.0028896
1.0000362
1.0000062
1.0000007
1.0000000
<EOF>
```

Figure 3 : Somple Density Profile

NB: The number of points in the density profile, NUMDEN, is linked implicitly with the depth of the ocean, DEPTH. (NUMDEN-1)/2 should divide DEPTH evenly i.e. (DEPTH modulo (NUMDEN-1)/2)) = 0

3. Output

The program produces an intermediate output file containing the integration data, but the main output file contains the velocity component data.

3.1. Velocity Component Output File

The velocity component output file contains a stream of six (6) columns of numbers which can be used by MATLAB (or other graphing utilities) to create several different type of graphs. Figure 4 shows a sample velocity component output file:

```
100.
             ٥.
         0.
                        0.00000E-00
                    0.
                                        -6.92253E-07
100.
        10. 0.
                    0. 7.92248E-04
                                      -5.79579E-07
100.
        20. 0.
                   0. 1.17258E-03
                                       -3.12313E-07
                   0. 1.118252
0. 8.22264E-04
100.
        30.
             0.
                       1.11829E-03
                                       -4.18162E-08
100.
        40.
             0.
                                        1.45645E-07
100.
       50. 0.
                                        2.32705E-07
                   0. 1.83084E-04
0. -1.63455E-05
100.
       60. 0.
                                        2.40433E-07
100.
       70.
             0.
                                        2.07768E-07
100.
       80.
             0.
                    0 -1.41567E-04
                                        1.55930E-07
       90.
            0.
100.
                    0 -2.05422E-04
                                        7.54386E-08
100.
      100. 0.
                    0. -2.11459E-04
                                       8.57241E-08
                   0.00000E-00
0.8.92248E-04
200.
        0.
            0.
                                       -6.96273E-07
200.
       10. 0.
                                       -3.76574E-07
200.
       20. 0.
                    0. 1.47278E-03
                                       -2.14316E-07
200.
                   0
       30. 0.
                        1.14799E-03
                                       -4.58662E-08
200.
       40.
            0.
                   0
                       8.52764E-04
                                        2.46645E-07
                   0.
       50. 0.
200.
                       4.70934E-04
                                        4.63655E-07
. . . .
       . . .
            . .
                   ٠.
. . . .
       . . .
                   . .
                          . . . . . . .
. . . .
       . . .
             . .
                   . .
                          . . . . . . .
1000.
       60.
             480.
                   0.
                       1.41223E-06
                                       -6.67499E-07
1000.
       70.
             480.
                   0.
                      1.42418E-06
                                       -8.10197E-07
```

Figure 4 : Sample Velocity Component Output File

It consists of 6 columns of data. The last three columns correspond to the integrals of U,V and W, respectively, at the point X,Y,Z specified in the first three columns. (NB: in this particular run of WAKE only the V and W velocity components were calculated, the U column therefore contains only zeroes). It is up to MATLAB or any other graphing utility to separate the data correctly and produce meaningful graphs. See Section 4: MATLAB.

Several different types of graphs can be produced. Figure 5 shows a 2-D Wave Plot and a 3-D mesh plot of the V Velocity Component taken at the ocean surface with the X-Y grid extending 6 and 2 kilometres respectively. Figure 6 shows a Matrix Intensity Plot, produced using PLOTZ, XV and MATLAB. Figure 7 shows the V and W Component Velocity Contour Plot. Figure 8 shows a zoomed in image of Figure 7 with arrows indicating the field flow.

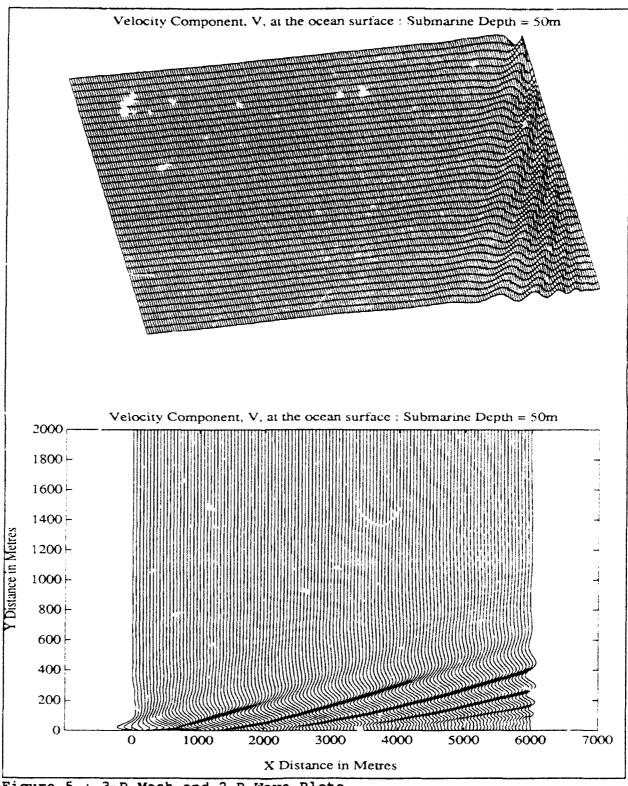


Figure 5: 3-D Mesh and 2-D Wave Plots

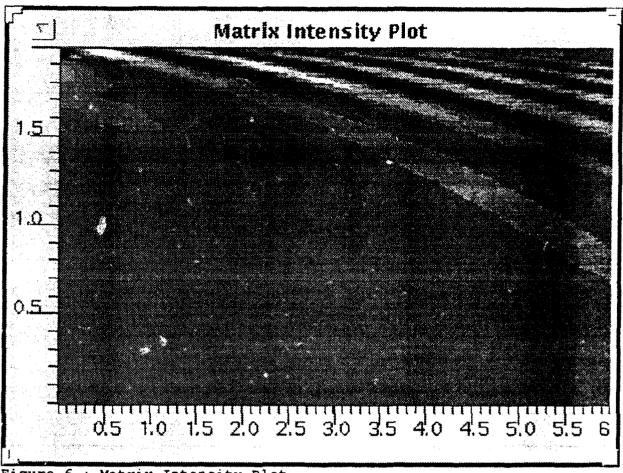


Figure 6 : Matrix Intensity Plot

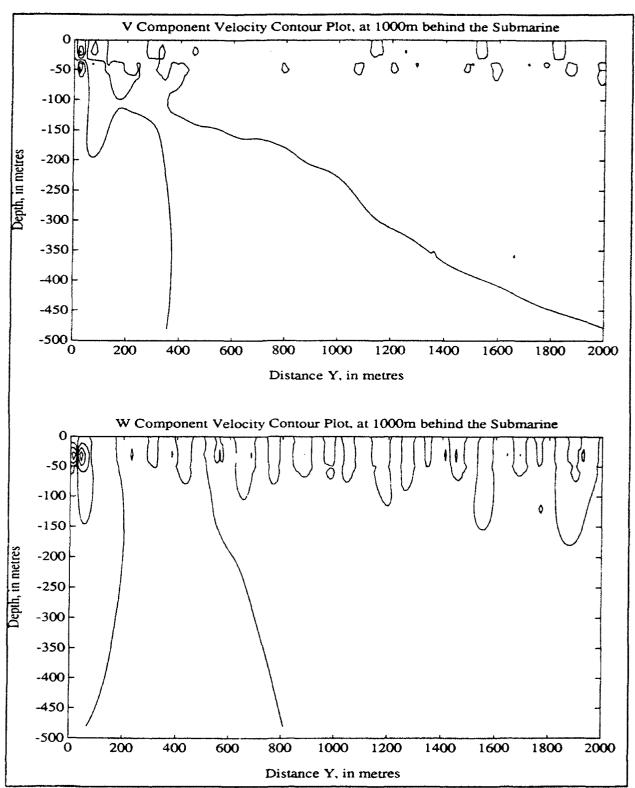


Figure 7 : V and W Contour Plots

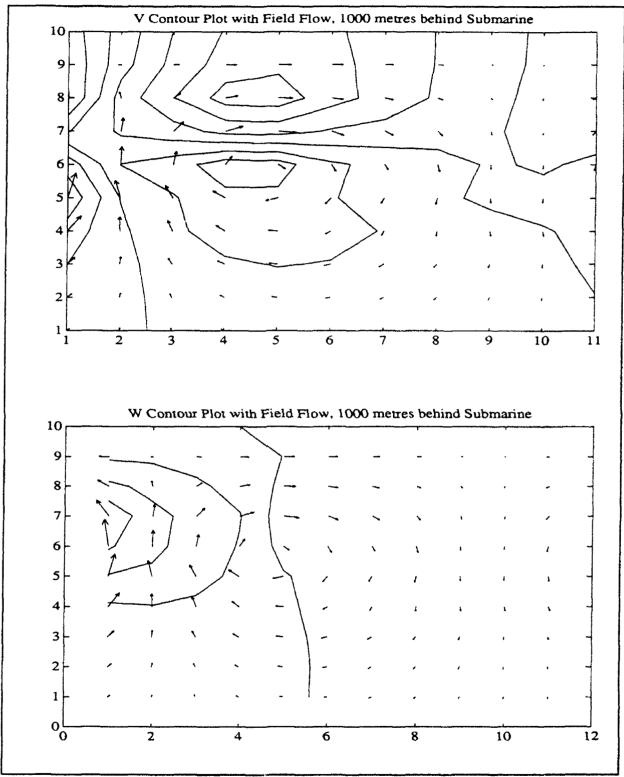


Figure 8 : Zoomed Image of Contour Plot with Field Flow Indicators

3.2. Integration File

The intermediate output produced by WAKE is the integration file. This file contains the data required for the integration. It consists of two parts, a Header and a Main Body.

Header : The header contains some of the parameters used in calculating the

amplitude functions. These are:

Mode Number, Number of Depth Values, Depth Step, Number of θ Values, Submarine Depth and Speed.

Main Body: The main body contains for each θ , k pair the amplitude functions

 $A(\theta)$ and $B(\theta)$ for all depth values.

Figure 9 shows a sample integration data file:

```
Mode Number
Number of 2 Values
HEADER
 Number of Theta Values
                                                                                         Contains the parameters
  30
Theta Step
2.05013822949197-02
                                                                                          used to generate the
Submarine Speed
2.50000000000000
                                                                                           Amplitude Functions
Theta
0.955754-5794732
 Wave Number, K
4.9903212012139D-18
    A(Theta)
-2.5419091178105D-20
                                B(Theta)
-2.5419091178105D-20
                                                              -480.000000000000
                                                                                                MAIN RODY
    -2.54137792918350-20
                                  2.1288060573866D-07
                                                              -470.000000000000
                                                              -460.000000000000
    -2.53977661974260-20
                                  4.2965042338244D-07
                                                                                        Repeated NUMTHETA times
    -2.53709788888995-20
                                                              -450.00000000000
                                                                                         from \theta_{min} = \pi/2 - \theta_{min}
   -3.2228172003193D-21
-1.5948747568649D-21
                                 3.2497013160336D-05
3.2650882233180D-05
                                                              3.45973986536620~23
                                                                0.0000000000000
 . . . . . . . . . .
Theta
1.5502949445000
 Wave Number, K
0.65904859979995
    A(Theta)
1.9842958911445-101
                                        B(Theta)
                                                              -480.00000000000
                                 1.9842958911445-101
5.5573161700118D-99
                                                              -470.00000000000
                                                              -460.000000000000
     6.0492250256296D-97
                                 1.1051914847512D-96
                                                              -30.000000000000
    -2.76223226063835-07
                                  2.4345912765042D-07
   -2.5742411595178D-09
-1.3120418273929D-11
4.2717012874524D-17
                                 4.0750962927742D-09
2.4372036970908D-11
2.4183772601375D-13
```

Figure 9 : Sample Integration Data File

4. MATLAB

To generate the graphs shown in Figures 5,6,7,8 requires a graphics package. The one described here is called MATLAB and can be run both on PC's and on SUN WorkStations.

To get started, invoke the matlab package with the command:

> matlab

After a moment, the MATLAB banner and a prompt like ">>" will appear. The MATLAB interpreter is awaiting instructions. Load in the velocity component data file generated by WAKE as follows:

>> load <filename>

After the file has been loaded, find out the size of the matrix that MATLAB has loaded the data into by executing the command:

>> whos

MATLAB will return with an answer that looks something like:

```
dolphin> matlab
                 < P R O - M A T L A B >
          (c) Copyright The MathWorks, Inc. 1984-1991
                   All Rights Reserved
                                18-Jul-1991
                Version 3.5i
              HELP, DEMO, INFO, and TERMINAL are available
>> load output.dat
>> whos
                                            Total
                                                      Complex
                 Name
                             Size
                           5015 by 6
                                            30090
                output
                                                         No
Grand total is (30090 * 8) = 240720 bytes,
>>
```

Figure 10 : Finding out about current variables and their sizes

Notice the matrix is a six column matrix. To generate diagrams like Figure 5 and Figure 6 requires reshaping the matrix into an appropriate format. This is done as follows:

```
>> u = reshape(filename(:,4),m,n);
>> v = reshape(filename(:,5),m,n);
>> w = reshape(filename(:,6),m,n);
```

where:

FILENAME = Name of the data file generated by WAKE without its file extension
The ":" symbol means "all". Thus, A(:,4) means extract all rows of
the fourth column of the matrix A, whereas A(4,:) means extract all
columns of the fourth row of the matrix A.

M = Number of Y-Grid points generated by WAKE

(i.e. M = ((YEND-YSTART)/YSTEP) + 1

N = A number. N is the number of X-Grid points generated by WAKE

(i.e. N = ((XEND-XSTART)/XSTEP) + 1

U = U is the resultant M x N matrix corresponding to the U

component velocity data (if COMPU = 'Y' ie WAKE has generated the data for the U velocity component)

V = V is the resultant M x N matrix corresponding to the V

component velocity data (if COMPV = 'Y' ie WAKE has generated the data for the V velocity component)

W = W is the resultant M x N matrix corresponding to the W

= W is the resultant M x N matrix corresponding to the W component velocity data (if COMPW = 'Y' ie WAKE has generated the data for the W velocity component)

4.1. 3-D Mesh Plots

To generate a 3-D Mesh Plot of the V Velocity Component as shown in Figure 5, execute the following command:

```
>> mesh(v,[hr,e])
```

where:

V = MxN matrix containing the V velocity component data

HR = Horizontal Rotation

E = Elevation

NB: A viewpoint matrix [HR,E] of [80,80] is recommended as a start.

A title may be put on the plot as follows:

```
>> title('3-D Mesh Plot')
```

4.2. 2-D Wave Plots

To generate a 2-D Wave Plot of the V Velocity Component as shown in Figure 5, execute the following commands:

First generate an offset matrix as follows:

$$\Rightarrow$$
 offset = ones(M,1)*linspace(0,TOP,N)/100000.;

where

N = Number of columns in our matrix V M = Number of rows in our matrix V

TOP = Arbitrary scale for the Y-Axis of the plot.

Next generate a scaling matrix as follows:

where

N = Number of columns in out matrix V

YMIN = Starting value of the Y-Axis YMAX = End value of the Y-Axis

Now define a new matrix, X which is the matrix V with each column offset slightly as follows:

$$>> x = v + offset$$
;

The 2-D Wave Plot, with an appropriate title, is generated in the Graphics Window with the following commands:

```
>> plot(x,scale,'b-')
>> title('2-D Wave Plot')
```

4.3. Matrix Intensity Plots

To generate a Matrix Intensity Plot of the V Velocity Component like Figure 6, requires a package called PLOTZ. First save the matrix A to a file. This is done by executing the command

This generates a file called WAVE.MAT, in the current directory, which contains the data from the matrix A. Now exit from MATLAB with the command

>> quit

To start up PLOTZ and generate the matrix intensity plot execute the following command:

> plotz wave.mat [1,2,3,4] xmin xmax ymin ymax

where

WAVE.MAT = File generated by MATLAB containing the velocity component data

[1,2,3,4] = Choose one of the following options:

1 = Grey
2 = Colour
3 = 4 Tones of Green
4 = Black & White (default)

XMIN = Starting value of the X-Axis

XMAX = End value of the X-Axis

YMIN = Starting value of the Y-AXIS

YMAX = End value of the Y-AXIS

4.4. Contour and Field Flow Plots

To generate the contour and quiver plots as shown in Figure 7 and Figure 8, run WAKE fixing X at some point behind the submarine, generating the V and W component velocities in the Y-Z Plane. First load the data and reshape the matrix into an appropriate format as follows:

```
>> load <filename>

>> v = reshape(filenam; ',5),m,n);

>> w = reshape(filename(:,6),m,n);
```

where:

FILENAME	= Name of the data file generated by WAKE without its file extension
M	= Number of Y-Grid points generated by WAKE
	(i.e. $M = ((YEND-YSTART)/YSTEP) + 1$
N	= A number, N is the number of Z-Grid points generated by WAKE
	(i.e. $N = ((ZEND-ZSTART)/ZSTEP) + 1$
V	= V is the resultant M x N matrix corresponding to the V component
	velocity data (if COMPV = 'Y' ie WAKE has generated the data for
	the V velocity component)
W	= W is the resultant M x N matrix corresponding to the W component
	velocity data (if COMPW = 'Y' ie WAKE has generated the data for
	the W velocity component)

4.4.1. Contour Plots

Set the scale on the Y and Z axis to correspond to the data generated.

```
>> yscale = linspace(ystart,yend,M);
>> zscale = linspace(-zend,zstart,N);
```

Plot and keep the empty axis as follows:

```
>> plot([ystart yend], [-zend zstart],'.')
>> hold on
```

Plot the V component velocity contour as follows:

```
>> vt = v';
>> contour(vt,yscale,zscale,'b-')
```

VT is now the transpose of the matrix v. This is required to get the correct orientation. Appropriate titles and axis labels are added with the following commands:

```
>> title('V Component Velocity Contour Plot, at 1000m behind the Submarine')
>> xlabel ('Distance Y, in metres)
>> ylabel ('Depth, in metres)
```

4.4.2. Field Flow Plots

To produce a contour plot with field flow arrows overlayed, display in the Graphics Window the contour plot and execute the following commands:

```
>> hold on
>> quiver(vt,wt)
```

where VT and WT are the transposes of the V and W velocity component matrices. This will produce a contour plot with field flow arrows overlayed. The arrows are scaled so that they indicate not only direction but magnitude. The scaling of the arrows become too small if we are looking at a wide area, hence it is necessary to look at small portions of the plot at any one time to see the structure of the field flow.

To look at a section of the plot like Figure 8 execute the following:

- >> clg >> hold off >> contour(vt(zindstart:zindend,yindstart:yindend),'b-')

- >> quiver(vt(zindstart:zindend,yindstart:yindend), wt(zindstart:zindend,yindstart:yindend))
- >> title('V Contour Plot with Field Flow)

Where:

ZINDSTART, ZINDEND, YINDSTART, YINDEND define the submatrix of the matrix VT.

This has the effect of zooming in on a portion of the plot so that we can see the structure of the field flow.

4.5. Printing

To print within MATLAB, display the graph required in the graphics window and then execute the following commands:

>> meta <filename>

This creates a file called FILENAME.MET in the current directory. Create a postscript file of the plot as follows:

>> !gpp filename.met -dps -ol

This generates a postscript file of the plot which can be sent to a postscript printer in the usual manner.

For more information on MATLAB and MATLAB commands consult the MATLAB user manual.

To print a matrix intensity plot like Figure 6, save the plot generated by PLOTZ, as follows:

- 1). Click on the plot with the right button of the mouse
- 2). Select Save to Raster
- Quit out of PLOTZ by clicking again on the plot with the right mouse button and 3). selecting Quit

PLOTZ has now generated a Raster file called "PLOTZ.RAS". To print this file convert it to PostScript format as follows:

1). Execute the Raster file viewing package XV as follows:

> xv plotz.ras

- 2). Click with the right button of the mouse on the plot. This brings up a menu with various options and parameters.
- 3). Select Save. This brings up another menu with more options.
- 4). Select PostScript and rename the output file (if desired) and click on OK
- 5). This brings up a final menu with all the PostScript options (i.e. portrait, landscape, paper size, etc). Select the parameter desired and click on **OK**
- 6). A rotating fish indicates that XV is converting the Raster file to a Postscript file.
- 7). Quit out of XV and send the PostScript file just generated to a PostScript printer in the usual manner.

5. Compiling

To compile the WAKE program the following files need to be copied into a working directory. Access to the IMSL library Routines DQDAG, CSAKM, CSVAL (fortran version) is also required.

MAKEFILE: Compiles all the routines into a library and links them all together.

Figure 12 shows a sample makefile.

COMP: A script file which compiles the source fortran files to object files and

creates/updates a library called WAKE which contains all the

necessary routines and functions. Figure 11 shows a sample compile

DENSITY.DAT Name of the file which contains the density profile.

WAKE.PAR Contains all of the parameters WAKE requires

WAKE.F Main Program controls all the following subroutines

BOUND.F Searches for changes of sign in the function $D(\theta,k)$ for a given θ and

saves the intervals in a matrix

COMPUTE A.F Computes the Amplitude Functions $A(\theta)$, $B(\theta)$.

COMPUTE_H.F Converts a real depth value into an integer grid point. DENSITY.F Reads in the density profile and computes MU(z).

DET.F Returns the value of $D(\sigma,k)$, where D = Determinant of the

Wronskian of the ODE.

EXPANALYT.F Calculates the analytical solutions for an exponential density profile. FINDINTERVAL.F Finds an interval within which a root of the function passed occurs.

FIND ZEROES.F Returns the $\sigma(0)$ values.

GETSIGVALS.F Uses a quadratic mapping technique to obtain an array of σ

values biased at the endpoints.

INIT.F Reads in the initial parameters from the parameter file. INTEGRATE.F Performs the integrations using the IMSL routine DQDAG.

LINEAR.F Performs a linear extrapolation to obtain a guess for the value of K at

the current σ value using two previous (σ,k) pairs.

LOOKUP.F Looks up the interval matrix to obtain the intervals where roots of the

function $D(\theta,k)$ occur.

READINT.F Reads in the integration data for a fixed depth.

Numerical Recipes Bisection Method. RTBIS.F RTSEC.F Numerical Recipes Secant Method. UINT.F Computes the U velocity integrand. VINT.F Computes the V velocity integrand.

WINT.F Computes the W velocity integrand.

WRITEHEAD.F Writes the header information to the integration file. WRITEINT.F Writes the integration data to the integration data file.

WFUNCNS.F Computes the functions W_1 , W_1 , W_2 , W_2 .

Contains all of WAKE's common variables. COMMON.BLK

Figure 11 : Sample Compilation File

```
wake: wake.o bound.o compute_a.o compute_h.b density.o det.o expanalyt.b findinterval.o tindrerbes b getsigval.o init o integrate o
linear.o lookup.o readint.o rtbis.o rtsec.o uint.o vint.o wint.o writenead.o writeint.o wiunones.o
         f77 -g wake.o -L. -lwake /home/shark/share/local/lib-imslib.a -o wake
                    bound.f
comp det wake
 bound.o:
                                                             Put path to IMSL Routines (fortran vesion) here ()
                    compute_a.f
comp compute_a waxe
 compute_a.o:
                    compute_h.f
comp compute_h wake
 compute_h.o:
 density.o:
                    density.f
                    det.f
romp det wake
det.o:
                    expanalyt.f comp expanalyt wake
 expanalyt.o:
find_zeroes.o:
                    find_zeroes.f
comp find_zeroes wake
                    getsigvals.f
getsigvals.o:
init.s:
                    init.f
                    integrate.f
comp integrate wake
integrate.o:
.. ---.0:
                    linear.f
                    lookup.f
comp lookup wake
lookup.o:
readint.o:
                    readint.f
comp readint wake
rtbis.o:
                    rtbis.f
                    rtsec.f
rtsec.o:
                    uinc.f
comp wint wake
uint.o:
vint.o:
                    vint.f
comp vint wake
                    wint.f
wint.o:
writehead.o:
                   writehead.f
writeint.o:
                    writeint f
                    comp writeint wake
wfunchs.o:
                   wfunchs.f
comp wfunchs wake
                   wake.f
f77 -g -c wake.f -o wake.o
wake.o:
```

Figure 12 : Sample MAKEFILE

6. Program Description

The program can be divided into four main stages:

STAGE 1: Initialisation

STAGE 2: Binding the roots of the function $D(\sigma,k)$

STAGE 3: Solve the O.D.E. and Calculate the Amplitude Functions $A(\theta)$, $B(\theta)$ STAGE 4: Form the 3 direction (x,y,z) integrands U,V,W and integrate them

These stages are described in the following sections.

6.1. Stage 1: Initialisation

Stage 1 sets up the initial conditions. Figure 13 describes diagrammatically the structure for Stage 1

```
READ IN THE DENSITY PROFILE. D c
AND CALCULATE µ c
WHERE c = DEPTH POSITION

FIND THE G VALUES FOR WAVE NUMBER. 4=2
FOR THE FILET "MAXMOD" INTERNAL WAVES
```

Figure 12 : Stage 1 :- Initialisation

The initial parameters are first read in from the parameter file, WAKE.PAR, see Section 2.1 for a description on the input parameter file. Next the program reads in from the file specified in the parameter, DENFIL, the density profile, see Section 2.2 for a description of the density profile.

6.2. Stage 2 : Binding the roots of $D(\theta,k)$

Once the initial parameters and density profile has been read in, consider the dispersion relation [see E.O. Tuck: Appendix 3 Submarine Internal Waves, 1992].

"Our concern here is very much with dispersion relations for internal waves, namely relations between wave speed c and wave number k. For convenience, we use instead of c a quantity proportional to its reciprocal square, namely

$$\sigma = \frac{g}{c^2}$$

where g is gravity Then we need a connection between k and σ , e.g. $k = K(\sigma)$. One of our first tasks is to compute this relation for a given density distribution." [EOT 92]

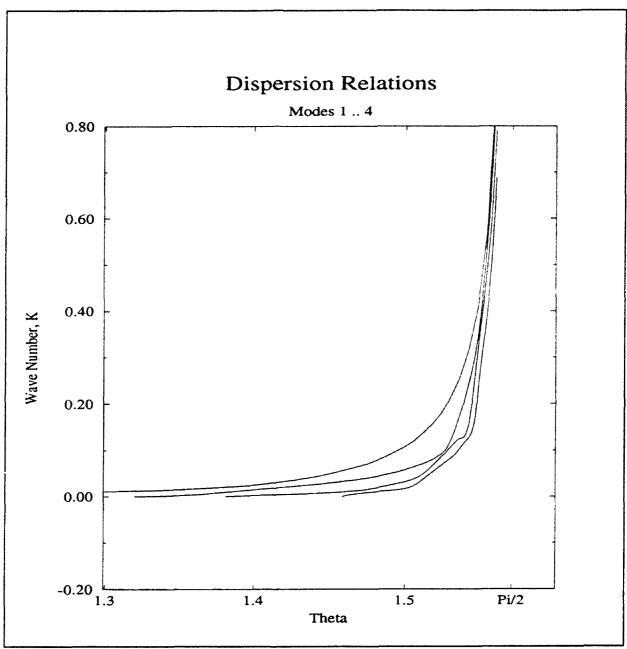


Figure 14: Dispersion Relations

Figure 14 shows the dispersion relation for the first 4 internal waves (The unseen line $\theta=k$ would correspond to the surface wave). We can see that as $\theta \to \pi/2$ the graphs get very close together and at $\theta = \pi/2$ there are in fact an infinite number of internal waves (this becomes a problem later when the program tries to pick out one particular internal wave and stay within that one mode).

Figure 15 shows diagrammatically the structure for Stage 2

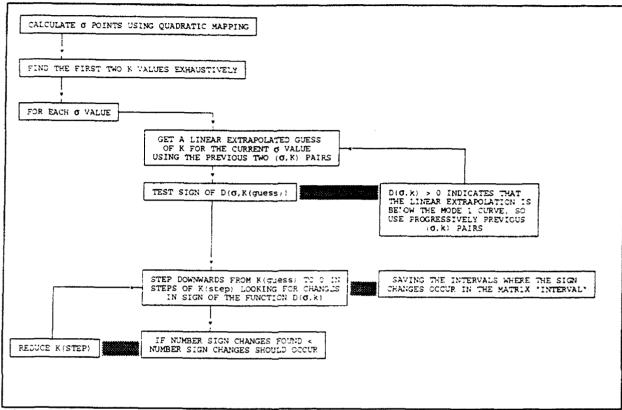


Figure 15 : Stage 4 : Binding the roots of $D(\theta, k)$

The dispersion relation needs to be accurately determined for multiple modes. The first step is to accurately determine the first mode as this acts as an upper bound on the other modes. The first mode is determined in the following manner:

Each mode has a θ_{min} , denoted θ^{mode} (e.g. the θ_{min} value for the first mode is denoted θ^1). Below this θ_{min} value internal waves do not exist for that mode (therefore below the θ_{min} value for the first mode, no internal waves exist). The interval θ^1 . THETAMAX is divided into NUMTHETA subintervals, each of size θ_{step} . The values of K for θ^1 and $\theta^1 + \theta_{step}$ are determine by an exhaustive search.

Once these first two values have been determined a linear extrapolation guess, K_{guess} , for the value of K at the next θ value can be obtained.

It was found that perturbations in the dispersion curve resulted in the linear extrapolation projected below the first mode curve. To overcome this a test on the sign of the function $D(\theta,k)$ at the point (θ_i,K_{guess}) is performed. If the function is negative then the extrapolation must be above the first mode, as desired (the probability of the extrapolation projecting into the third mode region where the function is also negative is considered to be negligible). If the function is positive then the extrapolation is below the first mode curve so the previous (θ,k) pair is used for the extrapolation. This process of using progressively previous (θ,k) pairs is repeated until a satisfactory linear extrapolated guess, K_{guess} has been obtained

The process is continued from $\theta^1 + \theta_{step}$ through to **THETAMAX**, at each step using the two previous (θ,k) pairs to obtain a linear extrapolated guess, K_{guess} , for the value of K at the next θ value. Once K_{guess} has been obtained, the program searches downwards in steps of K_{step} looking for intervals where a sign change in the function $D(\theta,k)$ occurs (i.e. searches for roots of the function $D(\theta,k)$). For any given θ value, the minimum number of sign changes that should occur is known through the determination of the θ^{mode} values. If the number of sign changes found is less than the number of sign changes that should occur then the K_{step} value used must have been too large, so K_{step} is halved and the search for sign changes is repeated until the correct number of sign changes that should occur have been found. Once the correct number of sign changes have been found the intervals are saved in a matrix with the columns corresponding to the various θ values and the rows containing the intervals where sign changes occur.

When the program wants to know the K value for a certain mode at a given θ value all that is required is to look up in the INTERVAL matrix to get the interval where the sign change occurs for the given mode and use the Bisection Method to hone in on the root. If a higher mode than the first mode is selected then the program still calculates NUMTHETA θ values between θ^{mode} and THETAMAX, however, 10 preliminary calculation are performed from θ^1 to θ^{mode} .

6.3. Stage 3: Solving the ODE

Figure 16 shows diagrammatically the structure for stage 3

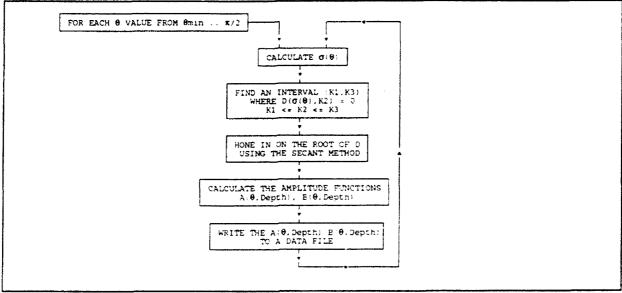


Figure 16: Stage 3:- Solving the ODE

In this stage we solve the ODE

$$(\rho W')' - (\kappa^2 \rho + \sigma \rho') W = 0$$

and calculate the amplitude functions $A(\theta)$, $B(\theta)$.

First we divide the interval θ_{mm} .. $\pi/2$ into NUMTHETA intervals and calculate θ_{step} ,

$$\theta_{step} = \frac{\frac{\pi}{2} - \theta_{min}}{NUMTHETA}$$

The Amplitude functions are then calculated at $\theta = \theta_{min}$.. $(\pi/2 - \theta_{step})$ in steps of θ_{step} . So for each θ we calculate the corresponding σ value.

$$\sigma = \kappa \sec^2(\theta)$$

By fixing σ and varying k we now find an interval $k_1...k_3$, where $D(\sigma,k_2)=0$ with $k_1 < k_2 < k_3$. Once this interval has been found (i.e. the root has been bounded) we use the Secant Method to hone in on the root. At this stage we have a $\theta(\sigma)$ and corresponding k pair, we use this pair to calculate the Amplitude functions $A(\theta)$, $B(\theta)$ as shown in Section 6.3.1.

6.3.1. Amplitude Functions $A(\theta)$, $B(\theta)$

For a detailed description of the method of calculating $A(\theta)$ and $B(\theta)$ see [E.O. Tuck 1992 : Submarine Internal Waves : Appendix 1]

NB: The amplitude functions are not defined explicitly in EOT 92, they appear implicitly in equation (4.1)

$$A(\theta) = \frac{kW_0}{\frac{\partial D(\theta, k)}{\partial k}} \quad ; \quad B(\theta) = \frac{W'_0}{\frac{\partial D(\theta, k)}{\partial k}}$$

where

$$W_{0}(z) = \begin{cases} -W_{2}^{J}(h) W_{1}(z) & : z \leq h \\ -W_{1}^{J}(h) W_{2}(z) & : h \leq z \leq 0 \end{cases}$$

and

h = Submarine Depth

 $W = W_1(z)$ and $W = W_2(z)$ are two separate solutions of the ODE

$$(\rho W')' - (k^2 \rho + \sigma \rho') W = 0$$

where

$$\sigma = \kappa \sec^2 \theta$$
; $\rho = Density$

Once the Amplitude Functions have been calculated the data is written to the data file specified in the parameter INTFIL as specified in the parameter file "WAKE.PAR".

6.4. Stage 4: Integrands and Integration

In this stage of the program we form the directional velocity component integrands U, V, Z and integrate them. Figure 17 shows diagrammatically the structure for Stage 4

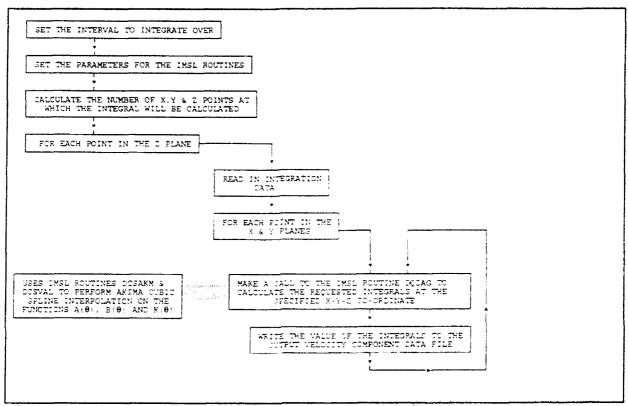


Figure 4: Stage 4:- Integrands and Integration

The integration is performed by the IMSL package routine DQDAG. This routine expects a continuous function (we achieve this by taking our discrete vectors which describe the functions $A(\theta)$, $B(\theta)$ and $K(\theta)$ and performing Akima Cubic Spline Interpolations, using IMSL routines DCSAKM and DCSVAL as required to obtain continuous smooth functions) and an interval to integrate over (in our case θ_{\min} . $\pi/2$). The IMSL integration routine then requires some desired accuracy and quadrature rules to be set. DQDAG is a general purpose integrator that uses a globally adaptive scheme in order to reduce the absolute error. It subdivides the integrating interval and uses a (2k+1)-point Gauss-Kronrod rule to estimate each subinterval. It was found that a Gauss-Kronrod Rule with 20 - 41 points, and a requested accuracy of 1.D-7 produced the best results.

The amplitude data $A(\theta)$, $B(\theta)$ is stored in the data file INTFIL. The amplitude data is read back into the program so that we have the arrays A, B containing the amplitude functions for all θ values at a specific depth. (The file INTFIL contains the amplitude functions for all depths at a specific θ value).

The first step is to define the Z-Grid over which the integrations will take place, then for each depth position (ie each point in the Z-Grid) the $A(\theta)$, $B(\theta)$ and $K(\theta)$ is read in. The next step is to define the X-Y Grid over which the integrations will take place. For each point in the X-Y-Z subspace the integration is calculated for each of the requested components and the results are written to the output velocity data file as described by the parameter **VELFIL**.

The U velocity component corresponds to the X direction (ie in the same direction that the submarine is moving). The V velocity corresponds to Y direction (ie sideways to the direction that the submarine is moving). The W velocity corresponds to the Z direction (ie the depth direction). The following three equations describe the integrals of the three velocity components.

$$\frac{w}{U} = -\frac{2V}{\pi} \int_{\theta_{\min}}^{\pi/2} \cos(kx \cos\theta) \cos(ky \sin\theta) \left[\frac{\sin(k \cos\theta L/2)}{L/2} \right] A(\theta) d\theta$$

$$\frac{u}{U} = \frac{2V}{\pi} \int_{\theta_{\min}}^{\pi/2} \sin(kx \cos\theta) \cos(ky \sin\theta) \left[\frac{\sin(k \cos\theta L/2)}{L/2} \right] B(\theta) \cos\theta d\theta$$

$$\frac{v}{U} = \frac{2V}{\pi} \int_{\theta_{\text{tild}}}^{\pi/2} \cos(kx \cos\theta) \sin(ky \sin\theta) \left[\frac{\sin(k \cos\theta L/2)}{L/2} \right] B(\theta) \sin\theta d\theta$$

where U = Submarine Speed

V = Submarine Volume

L = Submarine Length

X = Position in the X-Plane relative to the submarine

Y = Position in the Y-Plane relative to the submarine

 $A(\theta)$ = Amplitude Function, A

 $B(\theta)$ = Amplitude Function, B

k = Wave Number

7. Testing

Stage 1 of the program consists of reading in parameters and the density profile. This section can be tested by using the DBXTOOL debugging utility and stepping through each line of the code checking that the correct values are read in to the right variables.

Stage 2 of the program is concerned with binding the roots of the function $D(\theta,k)$. The method finally chosen (i.e. Linear extrapolated guess and then downwards search for changes in sign) was only the last of many techniques investigated, (others included searches using the secant method, fixing either K or θ). This final technique proved to be the most robust especially in terms of distinguishing the higher order modes. This section of the code was tested using the DBXTOOL debugging utility to ensure that the K value the program hones in on does indeed lie between the interval for that mode.

Stage 3 of the program is concerned with solving the ODE and computing the Amplitude Functions $A(\theta)$ and $B(\theta)$. For an exponential density stratified ocean, there exists exact analytical formulae for the functions W_1 , W_1' , W_2 , W_3' , the solutions of the ODE and the wronskian, D as follows:

Let

 $\lambda = \sqrt{disc}$

where

disc $= 4\delta\sigma - \delta^2 - 4k^2$ δ $= \ln(\text{Density Ratio})$, the natural logarithm of the ratio between the density of the ocean floor to the ocean surface k = Wave Number σ = Wave Speed Parameter If disc >= 0 then

$$W_{1} = e^{\frac{\delta}{2}(1+x)-k} \left[\left(\frac{2k-\delta}{\lambda} \right) \sin\left(\frac{\lambda(1+z)}{2} \right) + \cos\left(\frac{\lambda(1+z)}{2} \right) \right]$$

$$W'_{1} = \frac{\delta}{2}W_{1} + \frac{e^{\frac{\delta}{2}(1+z)-k}}{2} \left[(2k-\delta)\cos\left(\frac{\lambda(1+z)}{2}\right) - \lambda\sin\left(\frac{\lambda(1+z)}{2}\right) \right]$$

$$W_{2} = e^{\frac{\delta z}{2}} \left[\left(\frac{\delta-2\sigma}{\lambda}\right) \sin\left(\frac{\lambda z}{2}\right) - \cos\left(\frac{\lambda z}{2}\right) \right]$$

$$W'_{2} = \frac{\delta}{2}W_{2} + \frac{e^{\frac{\delta z}{2}}}{2} \left[(\delta-2\sigma)\cos\left(\frac{\lambda z}{2}\right) - \lambda\sin\left(\frac{\lambda z}{2}\right) \right]$$

$$D(k,\sigma,h) = (k-\sigma) e^{\delta(\frac{\lambda}{2}-h)-k} \left[\left(k+\frac{\delta}{2}\right) \frac{\sin\left(\frac{\lambda}{2}\right)}{\frac{\lambda}{2}} + \cos\left(\frac{\lambda}{2}\right) \right]$$

If disc < 0 then Lambda is a complex number so

$$\frac{\sin\left(\frac{\lambda}{2}\right)}{\left(\frac{\lambda}{2}\right)} - \frac{\sinh\left(\frac{\lambda}{2}\right)}{\left(\frac{\lambda}{2}\right)}$$

and

$$Cos(\frac{\lambda}{2}) \rightarrow Cosh(\frac{\lambda}{2})$$

where

$$\lambda = \sqrt{-4\mu\sigma + \mu^2 + 4k^2}$$

therefore our set of equations become:

$$\begin{split} W_1 &= e^{\frac{\delta}{2}(1+z)-k} \left[\left(\frac{2k-\delta}{\lambda} \right) Sinh \left(\frac{\lambda(1+z)}{2} \right) + Cosh \left(\frac{\lambda(1+z)}{2} \right) \right] \\ W_1' &= \frac{\delta}{2} W_1 + \frac{e^{\frac{\delta}{2}(1+z)-k}}{2} \left[(2k-\delta) Cosh \left(\frac{\lambda(1+z)}{2} \right) - \lambda Sinh \left(\frac{\lambda(1+z)}{2} \right) \right] \\ W_2 &= e^{\frac{\delta z}{2}} \left[\left(\frac{\delta-2\sigma}{\lambda} \right) Sinh \left(\frac{\lambda z}{2} \right) - Cosh \left(\frac{\lambda z}{2} \right) \right] \\ W_2' &= \frac{\delta}{2} W_2 + \frac{e^{\frac{\delta z}{2}}}{2} \left[(\delta-2\sigma) Cosh \left(\frac{\lambda z}{2} \right) - \lambda Sinh \left(\frac{\lambda z}{2} \right) \right] \\ D'(\lambda,\sigma,h) &= (k-\sigma) e^{\delta \left(\frac{1}{2}-b \right)-k} \left[\left(k + \frac{\delta}{2} \right) \frac{Sinh \left(\frac{\lambda}{2} \right)}{\frac{\lambda}{2}} + Cosh \left(\frac{\lambda}{2} \right) \right] \end{split}$$

These formulae are contained in the routine "EXPANALYTIC". If the program is run with an exponential density profile the program produces tables which can be used to verify that the analytic and numerical solutions agree. Figure 18 and Figure 19 are tables of data taken from the debugging file which show the numerical and analytic solutions for W_0 , W_1 , W_1' , W_2 , W_2' and the amplitude functions $A(\theta)$ and $B(\theta)$. These tables indicate that the numerical and analytic solutions agree to a satisfactory number of significant figures.

Internal Waves in Submarine Wakes

ľ	****	The second secon	
	,	2.45735953895490-03 \$[CMA 4 11.497315729739	
E	1 11) = 0 0074224000 M2 (0) + 2006 0515979445 0 6 0 03340659789 M2 (1) 6 2463 7285389922 1 6 0 02854803 M2 (2) 6 2063 488308424	
E	1 31	7	
ţ.	41	1 t 0 3410332455 M2 ; 51 s 2444.7247325145 1 t 0 3440325039 M2 ; 4) x 2497.3448353702	
-	1 73 1 9:	1	
	- 00 - f 10 - f 20 - f 30 - f 30 - 6 30 - 7 3 - 7	7	
E	111	; x = 0 3325740943 M2 (111 = 2748.3299490949 ; x = 0 3510158051 M2 (121 + 2736.1417859735 ; x = 0 3484997445 M2 (131 x 2734.5554479142	
E	1341	3 310134001 W	
ŧ;	15	: 2 1353899402 W2 (16) # 2613-742974941 - 3 1289717751 W2 (17) # 2565 1168675868	
ţ.	11	a 0 3239369549 M2 (30) x 2309 537049444 a 0 3239369646 M2 (19) a 2447 0945374345	
E:	22	s	
E	(23)	s 0 2000039000 W2 (22) s 2220 07933013131 0 0 2736151274 W2 (23) s 2132.0351315703 1 A 0 244603400 W2 (24) = 2016 1002407673	
E	(24)	1 c 3 2488779941 W2 (25) m 1949 0121314625 c 0.2354749214 W2 (24) m 1835 5491562027	
1	(24) (27) (28) (29)	0 + 0.2216354609 W2 (27) + 1726.0962025274 0 + 0.2067817063 W2 (28) + 1681.948668746	
Γ.	(32) (31)	0 1915859864 M2 (29) w 1893 4146334433 1 0 0 175888488 M3 (30) m 1370.8228633413	
Ė	(31)	7 x - 0 1996923887 MZ (31) x 1244.4984392443 A - 0.1830223399 MZ (32) x 1114.7893067378 C - 0.1946834734 MZ (33) - 982 0340834888	
E	(34)) n 0 1084108131 W2 (34) # 844.5303718382 4 0.0989429999 W2 (35) n 708 1099544874	
į.	(36)	# 0.0730279400 M2 (36) E 949.2391397704 4 0.0549131302 M2 (37) # 429.0574742637	
Ľ,	(34) (35) (36) (37) (30) (30)	0 03.036499830 M2 (38) w 285 4887344263 0 0382848092 M2 (38) w 142.5388825234	
1,	(46)	: 4 -0.0001293433 M2 (40) s -1.0000880000 : 2 - 0.0007594474 M2((-0) s - 3.8887478730 TEST s - 2.00001525	
E	1 1	0 x 0 000754478 M2'(0) x 0 0807470730 TEST x 0 00001325 0 x 0 0006844190 M2'(1) x 0 0382003411 TEST x 0 00001325 0 x 0 0006132375 M2'(2) x 4.7731360057 TEST x 0 00001325	
į.	1 3	5 f 0 0004123275 M2'(2) = 4.773136057 TEST = 0 00001525 1 T 0.0005381497 M2'(3) = 1 1950710177 TEST = 0 00001525 1 T 0.0004623472 M2'(4) = 3 4055940512 TEST = 0 00001525	
Ľ,	1.3	= 0.0003056655 M2 (5) = 3.0063026369 TEST = 0.00001526 = 0.0003077735 M7 (6) = 2.3988260175 TEST + 0.00001526	
1		5 0.001493689 M2'(7) = 1.7848397417 TEST = 0.00001526 5 0.001493689 M2'(8) = 1.1459286563 TEST = 0.00001326	
E	112	# 0.0000697692 42 1 91 4 0.7438628818 TEST 9 0.0001526 1-0.00001286 42 (201 1 -0.075667640 TEST 9 0.0001526 4 -0.0000969787 42 (11) 5 -0.7056553997 TEST 9 0.0001526	
E	: <u>::</u>	-3.30098999999 M2 (31) 5 -3 10303099 TEST # 0 0000324 -1 -0.002497246 M2 (31) 5 -1 3243348459 TEST # 0 0000324	
E.	111	2 -0 0805277812 92: [18] 1 -2 155018138 TEST 1 0 0000324 2 -0 00048248 92: [13] 4 -3 1540850723 TEST 2 0 0000324 4 -0 0004834828 92: [14] 4 -3 157050487 TEST 3 0 00001524	
Ľ,	1127	e -0 0004828641 M2:(16) 4 -5 "578030683 TEST 5 0 00001526 2 -0 0005575770 M2:(17) 5 -4 1647115317 TEST x 0 00001526	
E	133	= -0 0000311967 M2 (18) x -0 3199248742 TEST : 0 00003726 = -0 0007037226 M2 (18) x -0 4815795991 TEST = 0 00003726 = -0 0007037378 M2 (70) - 4 000448702 TEST = 0 00003726	
Ę,	121;		
E	123	# -a 0009072050 M2 (22) # - 0716265932 TEST > 0 00061526 - a 0009705110 M2 (23) # - 1678938967 TEST = 0 00061526 0 0010311809 M2 (24) # -8 0380161020 TEST = 0 00001526	
1	120		
ţ.,	(21:	; -3 0411957990 M2 (27) x -9 3209100482 TSCT x 0 00001524 2 -0 061249312 M2 (20) x -9 599375807 TSCT x 0 00001524 c -0 0612494827 M2 (20) x -10 0514857463 TSCT x 1.00001524	
	130	c -0 001201072 M2*(23) x -10 051037463 TEST x 1.00001526 -0 0013011070 M2*(50: 1 -10 0742077812 TEST x 0.00001526 	
	132:	: -0.001405332 M2 (32) : -10.3404799906 TEST : 0.2000337 0.001430401 M2 (33) : -11.3783449362 TEST : 0.2000337 0.0014404660 M2 (34) : -11.3783449362 TEST : 0.2000327 0.0014404660 M2 (34) : -11.378340900 TEST : 0.2000327	
ti.	(34)		
Ľ.	(36)		
E	077 (20) (10) (10) (10) (10) (10) (10) (10) (1	0 001349;971 02 131; 0 -10 072893100 TEST 0 00001331 0 001349321 02 131; 1 -10 1407100 TEST 0 00001331 0 001349321 02 131; 1 -10 1407100 TEST 0 00001331 0 001349301 02 131; 1 -10 14071000 TEST 0 0000137 0 001349301 02 131; 1 -10 14071034 TEST 0 0000137 0 0013493100 02 131; 1 -10 14071034 TEST 0 0000137 0 001349473 02 131; 1 -10 14071034 TEST 0 0000137 0 001349473 02 131; 1 -10 14071034 TEST 0 0000137 0 0013493104 02 131; 1 -10 14071034 TEST 0 0000137	
1			
t"		COMPUTE A	
-	ch of	2. 43739539894ED-95.5CRM r :: %6715725739 f Tolmmerine : 53.0002000000 onding grid point : # : 58 e. : 4481342050050-038 - EFE : 2.433192872248ED-03 :: .46711373538	
	EPS	* 2 4619242096013D-03K - EPS * 2.4931978722481D-02 11.347113725739 2.4573997389748D-031 1.5268243175337D-05	
!	1015	derivetive of D with respect to K. Ch = -2120,7891943131	
10	01 3 11 2 31 2 51 3 51 3 67 2 101 2 121 4 121 4		
Eo.	31	3 8544546407 A(2)x -0.000048442 B(2)x -0.0000034546 1 3 33463433468 A(3)x -0.000046442 B(3)x -0.00003202 2 4 0081334744 A(4)x -0.000046442 B(4)x -0.0000323958	
101		1 4.048495713 81 514 -0 0000847144 81 31 1 -0 0500021644 2 4 1182276310 81 814 -0 0000847148 81 81 2 -0 0000817270 4 118428183 81 772 -0 0000847148 81 771 -0 0000817270 4 118428183 81 772 -0 0000808182 81 771 4 -0 0000812870	
101	•	T + 1365628113 A(7) = -0.0000049162 B(7) T -0.0000012850 = -0.0000091745 A(8) = -0.0000048974 B(9) T -0.000008974 B(9) = -0.000008975 B(9) = -0.0000089715	
100	10:	z 4 1992671442 A(9):x -0 000004857 B(9) 2 -0 0000083735 z 4 2033196727 A(10):x -0 0000048708 B(10) x 0 000000374 z 4 1943443863 A(13):x -0 0000048623 B(11): 7 0000003562	
Ser.	12	# 1943#4943 A[1] # -0.00#0#4943 B[1] # 0.00#0#5942 7 4 1777#45443 A[1] # -0.00#0#49# B[1] # 0.00#0#594	
40	141	2 4 1479297598 A[13]: -0 UUGGUREGGE B[13]: U UUGGGE15995 : 4 1046178885 A[14]: -0 000047784 B[14]: 0 00000415195 : 4 0342421785 A[15]: -0 0000046977 B[15]: 0 0000022755	
10	14) t	7 99845335 A(16)	
Col	17 v 18 v	1 3317043022 A(18) = -0.000004358 B(18) : 0.000033421 2 3 7343923425 A(19) = -0.0000043294 B(19) x 0.0000039445	
Col	20; = 20; = 22; = 22; = 23; = 25; = 26; = 27; = 28; = 27; = 27; = 28; = 27; = 27; = 27; = 27; = 27; = 27; = 27; = 27; = 27; = 27; = 27; = 27; = 27; = 27; = 28; = 27; =	1 7 7 24 25 12 27 27 27 27 27 27 27 27 27 27 27 27 27	
E	1)	3 3906773226 A[22]4 -0 0800039288 B[22] 5 0 5000050913 7 3 256359409 A[23]4 -0 0800037734 B[23] 7 0 3000054444 7 3 1133792937 A[24] 6 00000516077 B[24] 7 0 00000578971	
10	251 2	.) 113579237 Alidis -0 commonidor7 Midai -0 6000877971 -2 18221371287 Alidis -0 commonidor7 Midai -0 600084718 -1 1012187287 Alidis -0 commonidor7 Midai -0 commonidor -1 1012187287 Alidis -0 commonidor7 Midai -0 commonidor -1 101218777 Alidis -0 commonidor Midai -0 commonidor -1 101218777 Alidis -0 commonidor -0 comm	
roi loi	27 2 28 2	1 1421357287 A1251w -0 3008044722 8(25) : 0.680684118 2 1 4018187078 A(26)x -0 30980)2474 8(26) : 0.009064200 2 1 4557155777 A(27) : 0.30980)2674 8(26) : 0.009864200 1 14612244227 A(26)x -0 0860828518 8(28) x 0.088064832	
101	29) : 30) :	: 1.2007429048 Al2912 -0.000036421 91291 0.0000072367 2.093636324 Al3019 -0.0000622292 Birol 0.0000072705 2.1003742195 Al3110 -0.0000622077 Bi31) 0.0000874840	
Eo;	32: 4	1 19091742145 A(31)4 -0 0000022017 B(31) 9 0.0000476840 9 1.7021260326 A(32)2 -0 0000019723 B(32) A 0.0000478767 9 1.499441862 A(32)2 -0 00000197378 B(3)3 9 0.0000088479	
Eo:	341 :	1. 14:244:274 Algir - 0 commonfield Hills - 0 commonfield - 0 co	
Le:	36 =	0 489798888 A131: - 0 060810871 8138; 2 3000882289 - 0 439798884 A1371: - 0 3000897372 B173; 2 0 3000887307 2 8 486189297 A1881: -0 3000895078 8128; 3 3,000887489 - 0 21748487820 A1391: -0 3000895078 8128; 3 0 0000866881	
201	27 22 27 27 27 27 27 27	0 0 0 0 0 0 0 0 0 0	
Γ"	•	0 d072558253 V(40; 0 0000000008 8[40; = 0 0000000120	
L			

Figure 18: Numerical Solution

Formula De Portura De	etio # 1 0019000000000 resmunant	
MI (91 s	0 3074224000 MQ (0) a 2394 3554200147	
1 1 1 1 1 1 1 1 1 1	0 3340459994 W2 (2' s 2043.7285944444 0 1238508943 W2 (2' s 2034 8292522522 0 346764453 W2 (3' s 2454 2417747534	
	0 3187408149 M2 (4) e 2625 0683947859 0 2618912478 M2 (5) e 2666 7247890037	
# 1 # 1	9 3440323066 HZ (6) a 3497 349213303 9 3492330005 HZ (7) x 2*22 2*24323058	
	0.3913952951 M2 (#) + 2739 98098044 0.3528219074 M2 (9) = 3750 241991144 0.3631788622 M2 (50) = 3767 637438622	
1 111	0 3525760999 W2 (11) : 2748,3309834509 0 3528760999 W2 (12) : 2748,3309834509 0 352876098 W2 (12) : 2756,1419937754	
41 (11) a	6 1686997500 M2 (35) a 2716 555887044 2 1656273372 M2 (14) = 2689,5658881453	
1 (13)	0.1e04149995 M2 (15) = 2459 2438452135 0.1953899932 M2 (14) = 2413.7461428134 0.14491793 M2 (17) = 2413.7461428134	
1 (10)	0 3219399976 W2 [18] A 3509 5171983124 0.5159308688 W2 [19] ± 2487,0944459288	
4 (31) :	0 3050692537 W2 (20) ± 2378 0388450907 0 2953782786 M2 (21) + 2302 6773919741	
1 (23)	0 28480378649 W2 (23) 4 2220 673481377 0 2736151177 W2 (23) 4 2122 6352506500 0 2616022498 W2 (24) 4 2019 1953745148	
(24)	3.248977933 M2 [23] = 1940.020223273 0.2354769199 M2 [26] = 1035.549242232#	
1 1271 +	0.206356387 W2 327; a 124,0963890009 0.2067917015 W2 528; x 1631 9487369205	
1301	G.1798581456 W2 (50) # 1570.8223454108 G.1798581456 W2 (50) # 1570.8223454108	
	20 20 20 20 20 20 20 20	
4 (3)	0.090429951 M2 (34) = 108 000435865 0.1084108091 M2 (34) = 844 630435865	
1 (37)	0.0738273133 W2 (34) 4 369 2991777807 0.0569131310 W2 (37) 4 428.0575804648 0.014448991 W2 (39) 4 287 4887346445	
1391 4	3 3182847894 H2 (39) = 162 3398923488 -0 0001293784 H2 (80) + -1 0000000000	
11.1 61 4	0 00075447 W[[] 0] 0 2 2887479823 0 000844019 W[] 1 1 1 3 312012703 0 000844019 W[] 1 1 1 3 312012703 0 00091819 W[] 1 1 1 3 512012703 0 00091819 W[] 2 1 1 1 12012124 0 00042347 W[] 2 1 1 1 12012124 0 00042347 W[] 2 1 1 1 12012124 0 00042349 W[] 3 1 1 12012124 0 000228911 W[] 4 1 1 12012124 0 00042124 W[] 5 1 12012124 0 00042124 W[] 5 1 12012124 0 00042124 W[] 5 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
	0 0 0 0 0 0 0 0 0 0	
12:1 41 t	2 0004625472 M2 (4) 1 3 405555488 2 0003856655 M2 (5) 4 3 4065031259	
1 7	2 0003077375 W2': 6, v 1 7908244997 c 030229432 W2': 1 1 1 7808102132	
7: 1	2 0000897652 M2': 9: 0 0 3434635280 -0 0000102294 M2':(0) : -0 07949635280	
1 1111	-4 0000901907 M2 (11) + 0 7050549459 -4 0001609146 M27(12) + -1 5245100550	
11 11 1	-0 0002e91909 M2'(10) : -1 9423944668 -0 0003277832 M2'(14) : -3 5139104727 -0 000464429 M2'(14) : - 444447287	
1 1172 7	1	
1111	-0 00000313647 M2*(18) + -4 9198245895 -0 0007032236 M2*(19) : -5 4819792444	
1 21	-< 0007733789 M2*(20) c -+ 0286479380 -v 00084(4047 M2*(20) c -+ 0587408000 -c 000803000 M2*(20) c -+ 0587408000	
1 (23)	-0 0010311805 M2:(23) 1 -7 3650897317 -0 0010311805 M2:(23) 1 -7 3650897317	
1 (26)	-0.0010890486 M2'(25) 1 =8 489090077 -0.0011439578 M2'(25) 4 -0.0171193961	
11(17) +	-0.0012997990 W2:[27] s -9.320909999 -0.0012843311 M2:[28] s -9.4993794837	
1.1301 :	-0 001289882 W2'(30) = -10 09189882 -0 0013891882 W2'(30) = -10 09189882 -0 0013891882 W2'(30) = -10 09189882	
111321 +	-0 0014035282 M21(32) # -10 9404798573 -0.0014340430 M21(5)) + -11 1783443231	
1 1351 4	-0 0014606406 M216361 v -01 3858309758 -0.0016833085 M216351 c -11,5623732680	
1 (37)	-0 001964594 WG (181 s -11 707891497 -0 001964594 WG (187 s -11 8207892799 -0 001964591 WG (181 s -11 801893581	
1 (39)	V 300041241 M2 [2]; 4 4 318731900 V 3000120180 M2 [2]; 6 4 5 918731900 V 3000170180 M2 [2]; 7 7 9 10 11424888 V 3000170180 M2 [2]; 8 1 1 10 11424888 V 3000170180 M2 [2]; 8 1 1 1142489000 V 3000180180 M2 [2]; 8 1 1 11424890000 V 3000180180 M2 [2]; 8 1 1 114248900000 V 3000180180 M2 [2]; 8 1 1 114248900000 V 3000180180 M2 [2]; 8 1 1 1142480000000000000000000000000000000000	
01 01 .	0 000754474 W 0 0 2 287747821 0 0 0 0 0 0 0 0 0	
21 21 2	1 +58928877 Af 61% -0 0000042397 Bf 6'c - < 0000042397 1 144004421 Af 12 - 0 0000042393 Bf 6'c - < 0000042393 Bf 6'c - 0 000004393 Bf 6'c - 0 000004239 Bf 6'c - 0 00000341 Bf 756293342 Af 718 - 0 0000034867 Bf 73 - = 000003297	
01 0) = 01 3	4 0001136949 A(4)= -0 000004444 B(4)= -0 000023940 4 040499988 A(5)= -0 00004714 B(5)= -0 000023645	
1 41 :		
1(1) 7	4 2003197128 A(10): -0 000004844 B(9): -0 0000090914 4 2003197128 A(10): -0 000048918 B(10): 2 0000090974	
	1	
0(13) s 0(16) T	1,14,14,14,14,14,14,14,14,14,14,14,14,14	
0 (14) x	3.9909433453 All61x -0.000046244 B134's 0.000027097 3.9145988214 A1171x -0.000045385 B1371x 0.0000031293	
11101 t	7 #317043309 A(18): -0 0000084401 B(18': 0 0000033424 1 7343823878 A(18): -0 000008127 B(14': 0 0000034468	
(21) v	2 ******** A:4418 -: UUDUUGECT7 B:20; 0 200008/696 3 3155790254 A:2134 0 000086739 B:23; 0 000086725 3 3906773214 A:22; 0 0000857291 B:22; 0 2000869917	
1231 -) 2545559438 A(23)# -8 0000037757 B(23)# 0 000055478 3 1135792900 A(24)# -0 0000034090 B(24)# 0 000057875	
(25) 4 (25) 4	2 962337180 A/251c -0 0000034327 B(251c 0 000064123 2 8034346898 A/241c -0 0000052677 B(241c 0 000064234 3 43445444 A/271c 0000053747 B(241c)	
0(27) x 0(28) e 0(29) •	4 NJPPATTIRE MIGITS OF COROCATORS WILLIAM OF COROCATORS AND ALLESS OF THE COROCATORS	
01301 2	2.612262897 A1281c -0 0000828520 B1281c -0 000084857 2.800282742 A1291 -0 000082443 B1271 -0 100007371 2.0590546715 A1001 -0 0000824234 B1371 -0 000087471 2.700174354 A1311 -0 0000822015 B1311 -0 000887471	
	1 7021259654 A1211a - C 0000015728 B1321v 0 0000097873 1 899444100 A1231a - C 0000021737 B1331v 0 00000904455 1 2924614040 A1231a - C 0000012473 B1341v 0 3000001873 0 8631704590 A1531a - C 0000012473 B1351v 0 3000001234 0 8631704590 A1531a - C 000000174 B1351v 0 30000001234 7 6333772727 A1771ja - C 0000001747 B1351v 0 30000000131	
21351 4	197444104 A[3]11 - 0 00000[177 01]337	
(37)	9 4535737737 A[37], -c coecos7774 B[37]; -c coecos7531 0 453210217 A[36]; -c coecos753 B[36]; -c cecos753 B[36]; -c	
::3 : 1 :	0 2176402803 A(391× -0 0000002522 \$(34): 0 000004604?	

Figure 19 : Analytic Solution

Stage 4 of the program forms and integrates the three velocity components U,V and W. To integrate these velocity components using the IMSL routine DQDAG requires continuous forms for the functions $A(\theta)$, $B(\theta)$ and $K(\theta)$. However $A(\theta)$, $B(\theta)$ and $K(\theta)$ exist only as discrete tabulated functions, therefore some form of interpolation was required.

Three different interpolation techniques were investigated. These were:

- 1). Linear Interpolation
- 2). Natural Cubic Spline Interpolation
- 3). Akima Cubic Spline Interpolation

Figure 20 shows the three different interpolation methods, used to produce a continuous function for $K(\theta)$.

The linear interpolation is quite good and would be satisfactory for $K(\theta)$, however for $A(\theta)$ and $B(\theta)$, it was suggested that a smoother interpolation technique would be required.

The next technique investigated was Natural Cubic Spline Interpolation. This technique had some unforseen side-effects. One of the properties of natural piece-wise cubic splines is that the first derivative at the end of one interval must equal the first derivative at the start of the next interval, however no attempt is made to preserve the function's shape and this can lead to some wild oscillations in certain circumstances as can be seen in the example in Figure 20.

The last technique investigated was Akima Cubic Spline Interpolation (IMSL routine DCSAKM). This routine is based on a method by Akima (1970) to combat wiggles in the interpolant. The result is that the shape of the curve produced by DCSAKM matches the shape of the data. This, clearly, is what is required and produces the best results of the three interpolation techniques.

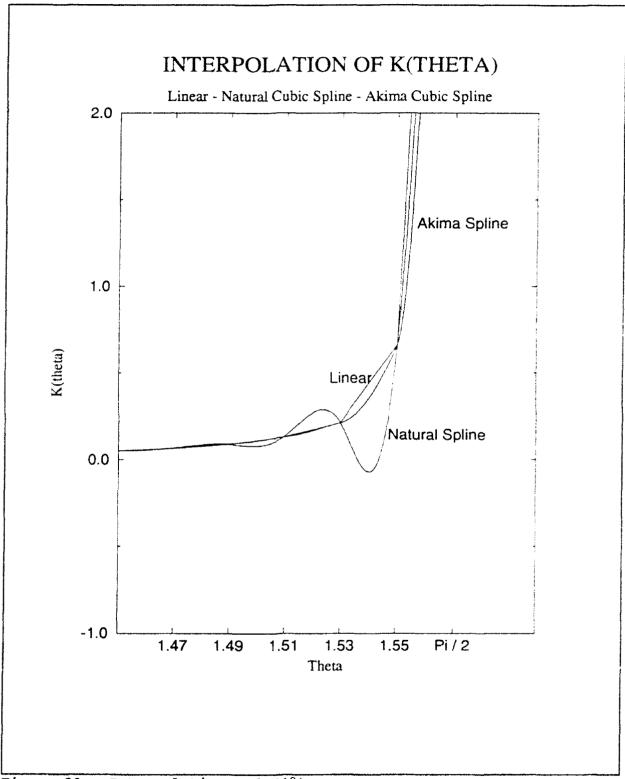


Figure 20 : Interpolations of $K(\theta)$

of

8. Error Messages

The following is a list of possible error messages

1100	Error opening parameter file "WAKE.PAR"
	The file "WAKE.PAR" must be in the same directory as the executable file
1101	Error occurred reading one of the parameters.
	This means that an error occurred reading the parameter file, possibly one of the parameters comment lines is not exactly 71 characters or the comment line contains tabs
1102	Invalid NUMTHETA in Parameter File (0 <= numtheta <= maxtheta) NUMTHETA must be odd.
	The number of theta values to calculate data for must be within the given range and must be odd.
1103	Invalid MODE in Parameter File (0 <= MODE <= maxmod).
	The mode to calculate data for must be within the given range.
1104	Invalid THETAMAX in Parameter File (THETAMAX < $\pi/2$).
	The maximum θ value allowed must be less than $\pi/2$. A value of 1.56 is recommended.
1105	Invalid SUBDTH in Parameter File (0 <= SUBDTH <= DEPTH).
	The depth of the submarine must be within the given range
1106	Invalid SUBSPD in Parameter File (0 <= SUBSPD <= maxspd).
	The speed of the submarine must be within the given range
1107	Invalid SUBLEN in Parameter File (0 <= SUBLEN <= maxlen).
	The length of the submarine must be within the given range
1108	Invalid SUBRAD in Parameter File (0 <= SUBRAD <= maxrad).
	The radius of the submarine must be within the given range

1109	Invalid XEND in Parameter File (XEND >= XSTART >=0)		
1110	Invalid XSTEP in Parameter File (XSTEP > 0).		
1111	Invalid YEND in Parameter File (YEND >= YSTART >=0)		
1112	Invalid YSTEP in Parameter File (YSTEP > 0).		
1113	Invalid ZSTART in Parameter File (ZSTART must be a multiple of "VALID GRID POINT"		
	All Z values must lie on the Z Grid as defined by the density profile. The Z Grid is defined by "(((NUMDEN-1)/2)+1)" Z values evenly spaced between 0 and DEPTH		
1114	Invalid ZEND in Parameter File (ZEND must be a multiple of "VALID GRID POINT"		
	All Z values must lie on the Z Grid as defined by the density profile. The Z Grid is defined by "(((NUMDEN-1)/2)+1)" Z values evenly spaced between 0 and DEPTH		
1115	Invalid ZEND in Parameter File (ZEND >= ZSTART >=0)		
1116	Invalid ZSTEP in Parameter File (ZSTEP must be a multiple of "VALID GRID POINT"		
	All Z values must lie on the Z Grid as defined by the density profile. The Z Grid is defined by "(((NUMDEN-1)/2)+1)" Z values evenly spaced between 0 and DEPTH		
1117	Invalid ZSTEP in Parameter File (ZSTEP >=0)		
1118	Invalid DEPTH in Parameter File (DEPTH > 0)		
	The depth of the ocean must be greater than zero		
1119	Invalid COMPU in Parameter File (COMPU = "Y" or "N").		
	Compute U Velocity Component, COMPU, must be "Y" for Yes!, or "N" for No!		
1120	Invalid COMPV in Parameter File (COMPV = "Y" or "N").		
	Compute V Velocity Component, COMPV, must be "Y" for Yes!, or "N" for No!		

1121	Invalid COMPW in Parameter File (COMPW = "Y" or "N").		
	Compute W Velocity Component, COMPW, must be "Y" for Yes!, or "N" for No!		
1200	Error occurred opening density profile file.		
	Either the file specified by the parameter DENFIL is missing or corrupt		
1201	Density profile file has a non odd number of points.		
	The density profile must have an odd number of points		
1202	Density Profile contains "X" points, not "NUMDEN" points as stated in the parameter file.		
	The parameter NUMDEN in the Parameter File should state the correct number of points that are contained in the density profile, DENFIL.		
1300	Error opening debugging file.		
1400	Error opening velocity component file.		
1500	Error opening intermediate integration file		
1501	Error occurred writing $A(\theta)$ data to the integration file		
	Either the integration file is missing or it is corrupt		
1700	Negative Sigma Value passed to Function LOOKUP.		
1701	Sigma > Sigarray(MAXMOD) in Function LOOKUP.		

The sigma value passed to the function lookup is greater than the maximum sigma value allowed, (i.e. the Sigma value for K=0 for the maximum mode calculated (mode 30)).

9. Related Documents

The following documents have been used as references or may be used as further reading.

- SUBMARINE INTERNAL WAVES Professor Ernie Tuck, 1992 Department of Applied Mathematics University of Adelaide, S.A. Australia

- INTERNAL WAVES RADIATED BY A MOVING SOURCE Volume 1. Analytic Simulation Michael Milder, 1974 R and D Associates Office of Naval Research Advanced Research Projects Agency

- SYNTHETIC APERTURE RADAR IMAGING OF SHIP GENERATED INTERNAL WAVES

Gasparovic, Thompson, Apel 1987 John Hopkins APL Technical Digest Volume 10, Number 4, 1989

- THE DREP INTERNAL WAVE NORMAL MODE MODEL-THEORETICAL BACKGROUND T.W. Dawson, April 1988 Technical Memorandum 88-7

SECURITY CLASSIFICATION	OF	THISTAGE

UNCLASSIFIED

REPORT NO. MRL-GD-0050	ar no ar-008-257	REPORT SECURITY CLASSIFICATION Unclassified
TITLE		
The WAKE software suite - a p	program to predict the internal density stratified ocean:	waves generated by a moving subsurface body in a User's guide
AUTHOR(S) M.A. Carroll		CORPORATE AUTHOR DSTO Materials Research Laboratory PO Box 50 Ascot Vale Victoria 3032
REPORT DATE January, 1993	TASK NO NAV 92/040	sponsor RAN
FILE NO G6/4/8-4453	REFERENCES	PAGES 41
CLASSIFICATION/LIMITATION REV	TEW DATE	CLASSIFICATION RELEASE AUTHORITY Chief, Maritime Operations Division
SECONDARY DISTRIBUTION		
	Approved for publi	ic release
ANNOUNCEMENT		
	Announcement of this rep	ort is unlimited
KEYWORDS		
Density Profiles Velocity Components	Computer Modelling 2D Wave Plots	Contour Plots
ABSTRACT		

This report provides details of the requirements needed to run the computer program WAKE. WAKE was developed during the period February-March 1992 as a result of a research project which was undertaken within DSTO Salisbury. WAKE calculates the internal wave velocity field generated by a moving subsurface body in a density stratified ocean.

M.A. Carroll

(MRL-GD-0050)

DISTRIBUTION LIST

Director, MRL
Chief, Maritime Operations Division
Research Leader, Airborne Sonar
M.A. Carroll, EBOR Computing
MRL Information Service

Chief Defence Scientist (for CDS, FASSP, ASSCM) (1 copy only)

Director, Surveillance Research Laboratory

Director (for Library), Aeronautical Research Laboratory

Director, Electronics Research Laboratory

Head, Information Centre, Defence Intelligence Organisation

OIC Technical Reports Centre, Defence Central Library

Officer in Charge, Document Exchange Centre (8 copies)

Army Scientific Adviser, Russell Offices

Air Force Scientific Adviser, Russell Offices

Navy Scientific Adviser, Russell Offices - data sheet only

Scientific Adviser, Defence Central

Director-General Force Development (Land)

Senior Librarian, Main Library DSTOS

Librarian, MRL Sydnev

Librarian, H Block

UK/USA/CAN ABCA Armies Standardisation Rep. c - DGAT (8 copies)

Librarian, Australian Defence Force Academy

Counsellor, Defence Science, Embassy of Australia - data sheet only

Counsellor, Defence Science, Australian High Commission - data sheet only

Scientific Adviser to DSTC, C/- Defence Adviser - data sheet only

Scientific Adviser to MRDC, C/- Defence Attache - data sheet only

Head of Staff, British Defence Research and Supply Staff (Australia)

NASA Senior Scientific Representative in Australia

INSPEC: Acquisitions Section Institution of Electrical Engineers

Head Librarian, Australian Nuclear Science and Technology Organisation

Senior Librarian, Hargrave Library, Monash University

Library - Exchange Desk, National Institute of Standards and Technology, US

Exchange Section, British Library Document Supply Centre

Periodicals Recording Section, Science Reference and Information Service, UK

Library, Chemical Abstracts Reference Service

Engineering Societies Library, US

Documents Librarian, The Center for Research Libraries, US

Dr D. Richardson, MOD

Dr J. Ternan, MOD

Dr A. Theobald, MOD

Dr J. Vrbancich, MOD

Dr J. Smelt, MOD

Dr R. Webster, MOD

Dr G. Furnell, MOD

Chief, Microwave Radar Division, SRL

Chief, Optoelectronics Division, SRL

Research Leader, Surveillance Systems, SRL

Dr G. Haack, MRD, SRL

Dr C. Anderson, MRD, SRL

Dr D. Madurasinghe, MRD, SRL

Dr D. McDonald, MRD, SRL

Mr W. Cumpston, EBOR Computing